(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 221 303**
A2

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 86112765.2

(51) Int. Cl.⁴: G06F 12/06

(22) Date of filing: 16.09.86

(30) Priority: 28.10.85 US 791884

(43) Date of publication of application:
13.05.87 Bulletin 87/20

(84) Designated Contracting States:
DE FR GB IT

(71) Applicant: International Business Machines Corporation
Old Orchard Road
Armonk, N.Y. 10504(US)

(72) Inventor: Aakre, David Elgar
Rt. 2, Box 22
Kasson MN 55944(US)
Inventor: Abrahamson, Douglas John
612 NW 29 Street
Rochester, MN 55901(US)
Inventor: Berglund, Neil Clair
Rt. 1, Box 144
Kasson, MN 55944(US)
Inventor: Bettermann, Vincent Alfred
2806 NW 15 Avenue
Rochester, MN 55901(US)
Inventor: Heim, Lloyd Stephen
Rt. 1
Oronoco, MN 55960(US)
Inventor: Moe, Kenneth Ardelle
15547 Linda Avenue
Los Gatos California 95030(US)

(74) Representative: Vekemans, André
Compagnie IBM France Département de Propriété Intellectuelle
F-06610 La Gaude(FR)

(54) Automatic I/O address assignment.

(57) An automatic address assignment system (10) has a plurality of I/O devices (20,22,...M,28,30,...P(14) coupled to a bus. Each device contains a unique machine-readable identifier which is used to select the device for address assignment. The identifier is a binary bit string. Each bit position in the bit string is selected by the host in a serial manner with the host specifying which binary value is being solicited. All devices whose identifier digit matches the solicited value respond positively and remain in contention for address assignment. The other devices will not respond and drop out of contention for address assignment until the sequence is restarted from the first bit. After the bit sequence is completed, the address for that device is bused to the device, and the sequence is restarted from the first bit until all devices have been assigned an address.

AUTOMATIC I/O ADDRESS ASSIGNMENT

This invention relates to the assignment of addresses to devices connected to a system by a bus, and in particular, automatic assignment of the addresses by the system in an efficient manner.

In computer systems, bus attached devices such as tapes, disk drives, monitors and other I/O devices require an address in order to be selected and used by the system. Typically, the address is predetermined and fixed by either physical location on the bus or by manual setting of switches or jumper wires. Many standard buses are not architected to use physical location to determine addresses, and manual setting of switches or jumper wires is prone to error even when performed by trained service personnel. There has been no efficient manner to accurately assign addresses without service personnel or operators actually taking the time to set the addresses for each I/O device. This is undesirable when systems are moving more toward customer setup and greater ease of use.

U S -A-4,360,870 describes a central processing unit which assigns addresses to I/O devices by first describing a device type and then assigning the addresses as a function of priority as established by the I/O devices of that type. This assignment scheme is dependent upon an established priority within a type of I/O device. It also requires the customer to have knowledge as to what types of devices are attached and to establish a priority for each I/O device.

IBM Technical Disclosure Bulletin, Vol. 24, No. 7B, December 1981, " Programmable Assignment of Device Addresses", discloses a method of assigning addresses where the I/O devices are serially connected by a line. The addresses are assigned as a function of a priority wiring scheme and serial propagation. These methods are not available for use with many standard busses without undesirable alteration of the standard bus.

An automatic address assignment system has a plurality of I/O devices coupled to a bus. Each I/O device has a unique fixed length identifier and is capable of responding to information on the bus as a function of the identifier. The system selects the first bit in the identifier and transmits a binary value, '0' or '1' to the I/O devices. All devices having identifiers with a first bit matching the transmitted binary value respond. If no devices respond, the system transmits the opposite binary value for the selected identifier bit. If one or more devices respond, the system selects the next identifier bit and transmits a binary value again. Those devices that did not respond to the previously selected bit drop out of the assignment process until it is restarted with the selection of the first identifier bit. As successive identifier bits are selected, more I/O devices drop out of the assignment process until only one device remains after the last bit is selected, at which time the I/O device is given an address and does not respond to subsequent assignment sequences. The process is repeated starting with the first bit until all I/O devices have responded and have been assigned addresses.

The automatic address assignment does not require a user to set any switches, or place an I/O device in any predetermined sequence on installation. An I/O device can be added at any time and will be automatically assigned an address.

Since devices drop out of contention once they have been assigned an address, the assignment is very efficient. One device will be selected and assigned an address on each assignment sequence. Assignment sequences are repeated until no device responds to either transmitted binary value when the first identifier bit is selected.

Fig. 1 is a block diagram of a host processor coupled to peripheral devices to be assigned addresses in accordance with the present invention.

Fig. 2 is a block diagram of a controller for each peripheral device which facilitates automatic address assignment by the host processor of Fig. 1.

Fig. 3 is a timing diagram showing a bit query and elimination process of the automatic address assignment system of the present invention.

A preferred embodiment of a computer system having automatic address assignment for its peripheral devices is indicated generally at 10 in Fig. 1. The system 10 comprises a host processor unit 12 coupled by a communicative means such as bus 14 to a plurality of input/output processors 16, 18,...N. I/O processor 16 is in turn coupled to peripheral devices 20, 22,...M by a communicative means such as a device bus 26. I/O processor 18 is also coupled to peripheral devices 28, 30,...P by a similar second device bus 32. Automatic address assignment is accomplished by I/O processors 16, 18,...N in operation with their corresponding peripheral devices. Host 12 could also use automatic assignment to assign addresses to I/O processors 16, 18,...N if desired. Device busses 26 and 32 preferably comprise parallel buses having 24 signal lines:

One interrupt line: ATTENTION.

BUS B (bit P)	RESPONSE IN
BUS A (bit 0)	DATA OUT
BUS A (bit 3)	DATA OUT
BUS A (bit 6)	DATA OUT
BUS A (bit 1)	ENABLE OUT
BUS A (bit 4)	ENABLE OUT
BUS A (bit 7)	ENABLE OUT
BUS A (bit 2)	CLOCK OUT
BUS A (bit 5)	CLOCK OUT
BUS A (bit P)	CLOCK OUT

TABLE 1

Each peripheral device comprises a device functional controller (DFC) which is indicated generally at 40 in Fig. 2 and controls communication between IOP 16 and peripheral device 20 in Fig. 1. Device bus 26 is coupled to drivers and receivers indicated at 42. The drivers and receivers both provide and detect signals on the individual lines of device bus 26 to enable communication. A link adapter 44 is coupled to the drivers/receivers 42 by a bus 46. Link adapter 44 contains control logic for executing device bus 26 protocols, and in normal operation, contains the device bus address so that bus selection and information transfer can occur.

Information is transferred across the device bus between the IOP and the DFC utilizing a random access memory 48 which is coupled to the link adapter 44 by a bus 50. A microprocessor 52 is also coupled to bus 50 and controls information transfer to and from memory 48.

When initially starting the system, or after new devices are added to the system, each device that requires an address activates its 'Attention In' line. The system then sets the three outbound tag lines to a master reset state. A maintenance mode facility indicated at 56 becomes active and a maintenance mode shift register 58 becomes ready to receive maintenance mode commands. Register 58 is coupled to link adapter 44 by a line 59, and receives the maintenance mode commands from link adapter 44. The maintenance mode facility 56 is coupled to register 58 by a bus 62 to receive the commands.

A non-volatile memory 60 is coupled to line 50 and contains a unique identifier for the peripheral device 20. Each peripheral device contains a unique machine-readable identifier in its non-volatile memory. The unique identifier is preferably a 48 bit binary string defining the manufacturer, unit type, and serial number of the peripheral device.

In maintenance mode and when the command to perform address assignment is activated, IOP 16 selects each identifier bit position, one at a time, and the maintenance mode facility 56 accesses the identifier bit string from the non-volatile memory 60. The maintenance mode facility compares the identifier bit value with the value solicited by the IOP and either responds positively, if the values match, or drops out of automatic addressing mode if the values do not match. If the maintenance mode facility responds positively to all bits in the unique identifier, it remains the only facility still selected and the IOP sends the command to shift the assigned address into the maintenance shift register. The maintenance mode facility then moves the assigned address into the link adapter logic and the automatic address assignment is completed for the device.

Once the automatic address assignment has occurred, the maintenance mode facility in the DFC 40 will no longer respond to the commands for automatic addressing unless the assigned address is reset. This allows the IOP to assign each device a unique address by repeating the assignment operation, once for each device. Addresses can be changed by the IOP. The IOP first selects the device using the currently assigned address. Then a write to address register with the changed address accomplishes the change. Devices that have only fixed addresses ignore this command.

Once Enable Out is dropped by the IOP, the device that remains isolated after the bit query sequence must assume that it is selected and accept the next order as a command. The IOP indicates a write sequence, raises Enable Out, and passes a Source/Sink order to the selected device. The Source/Sink order indicates the address register as the destination for the bus address. Address '0000'B is assumed by all automatically assignable devices as the address of the device address register. The Enable Out line is dropped, another write sequence is indicated, and a Data Transfer order is sent to the device. The Data Transfer order contains a 3 bit device address which is written into the device address register. After Enable Out is dropped, the device receiving the address no longer reacts to the address assignment command. The IOP then continues to address the other devices that do not have addresses, or stop the process if all devices are addressed. The IOP determines if any more devices need address assignment by doing another power poll and looking for a parity bit returned.

If at any time, the IOP determines that the address of any of the devices needs to be changed, maintenance mode is entered, and only the device which needs to change will be selected. At that time, a new address is assigned using the Source/Sink order, and the Data Transfer order.

While the above invention has been described with reference to preferred embodiments, it is recognized by those skilled in the art that further embodiments are within the scope of the invention. Configuration of the bus may be different. The addressing scheme could also be modified such that when a single device responds to the bit query, it is immediately assigned an address. Further intelligence built into the process could keep track of bit patterns unsuccessfully tried before, so as not to repeat unnecessary patterns.

In a further embodiment, after a predetermined number of identification bits have been transmitted, each remaining device transmits its unique identifier on the bus and receives from the bus at the same time. If the received data at a device does not match its transmitted identifier, the device raises the parity bit to indicate that more than one device remains and to continue the bit query sequence. If the parity bit is not raised, there is only one device in contention and it is assigned an address.

Claims

1. A method for automatically assigning addresses to a plurality of I/O devices communicatively coupled to a controller, each I/O device having a unique identifier and being capable of responding to communicated information as a function of the identifier, characterized in that it comprises the steps of:

(a) communicating identification data from the controller to the I/O devices;

(b) receiving responses from I/O devices having unique identifiers, at least a beginning portion of which matches the identification data received; and

(c) assigning an address to an I/O device if said I/O device is the only I/O device which responded to the identification data transmitted.

2. The method of Claim 1 further characterized in that it comprises the step of:

(d) starting at step (a) again with further identification data if more than one I/O device responded.

3. The method of Claim 2 characterized in that addresses are assigned to an I/O device when the identification data completely matches the unique identifier of an I/O device.

4. The method of Claim 2 characterized in that it further comprises the steps of:

(e) determining if all I/O devices have been assigned an address;

(f) starting at step (a) again, if all I/O devices do not have an assigned address; and

(g) inhibiting devices from responding if said devices already have an address.

5. The method of any one of Claims 1-4 characterized in that only one bit of identification data is transmitted in the first step (a), and only one bit of identification data is added in each step (d).

6. The method of Claim 5 characterized in that I/O devices which did not have identifiers matching each bit transmitted identification data drop out of contention for an address until the first bit position of identification data is transmitted again.

7. The method of Claim 4 characterized in that the unique identifier comprises a plurality of bits, and wherein the identification data also comprises a plurality of bits.

8. The method of Claim 1 characterized in that at least one I/O device first informs the controller that it does not have an address assigned.

9. The method of Claim 8 characterized in that I/O devices having an address assigned do not respond to identification data.

10. The method of Claim 1 characterized in that each I/O device has a non-volatile memory containing the unique identifier.

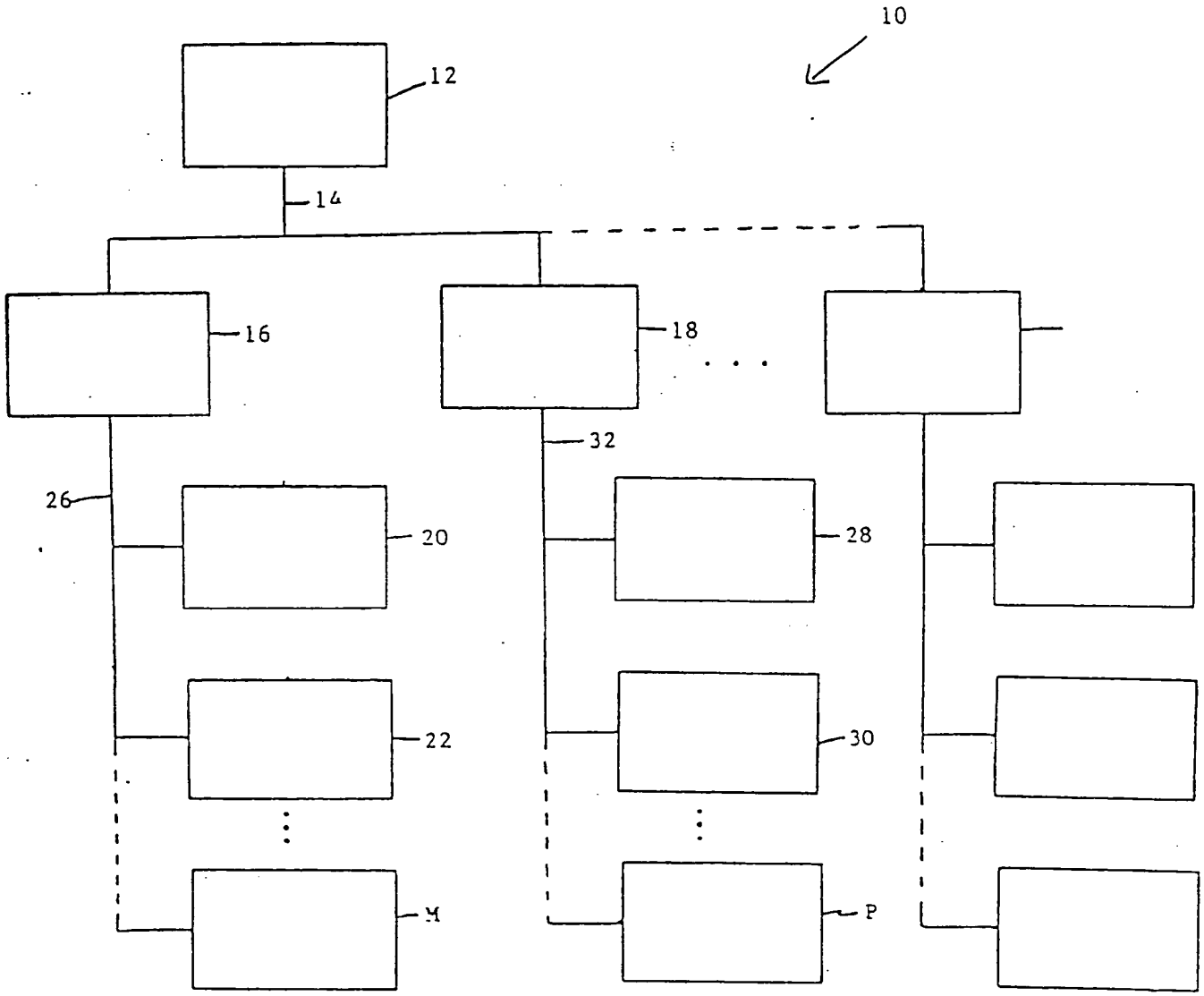

FIG. 1

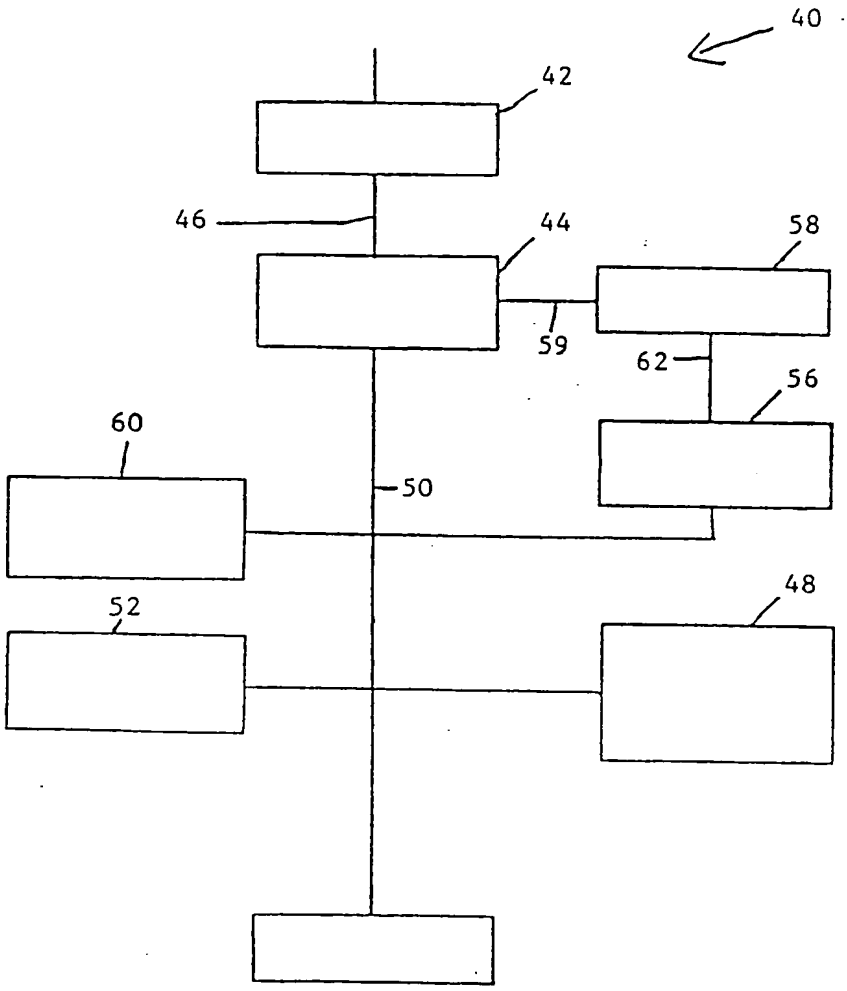

FIG. 2

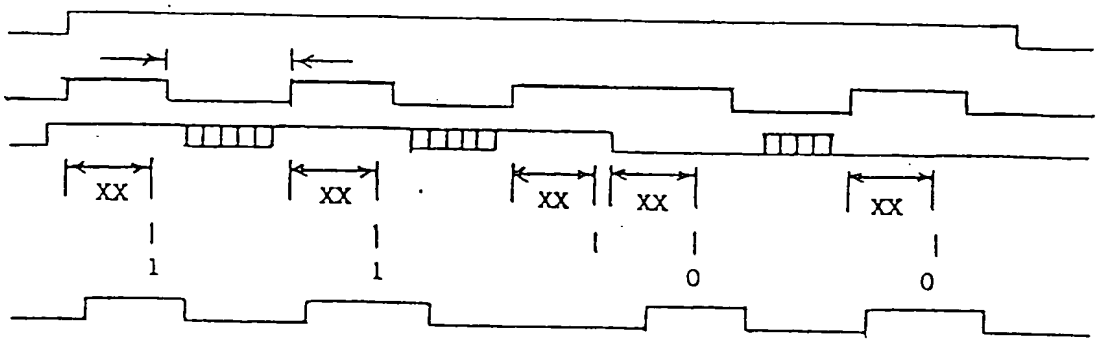

FIG. 3

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 221 303**
A3

(12) EUROPEAN PATENT APPLICATION

(21) Application number: **86112765.2**

(51) Int. Cl.⁴: **G06F 12/06**

(22) Date of filing: **16.09.86**

(30) Priority: **28.10.85 US 791884**

(43) Date of publication of application:
13.05.87 Bulletin 87/20

(84) Designated Contracting States:
DE FR GB IT

(88) Date of deferred publication of the search report:
18.10.89 Bulletin 89/42

(71) Applicant: **International Business Machines Corporation**
Old Orchard Road
Armonk, N.Y. 10504(US)

(72) Inventor: **Aakre, David Elgar**
Rt. 2, Box 22
Kasson MN 55944(US)
Inventor: **Abrahamson, Douglas John**
612 NW 29 Street
Rochester, MN 55901(US)
Inventor: **Berglund, Neil Clair**
Rt. 1, Box 144
Kasson, MN 55944(US)
Inventor: **Bettermann, Vincent Alfred**
2806 NW 15 Avenue
Rochester, MN 55901(US)
Inventor: **Heim, Lloyd Stephen**
Rt. 1
Oronoco, MN 55960(US)
Inventor: **Moe, Kenneth Ardelle**
15547 Linda Avenue
Los Gatos California 95030(US)

(74) Representative: **Vekemans, André**
Compagnie IBM France Département de Propriété Intellectuelle
F-06610 La Gaude(FR)

(54) Automatic I/O address assignment.

(57) An automatic address assignment system (10) has a plurality of I/O devices (20,22,...M,28,30,...P-(14) coupled to a bus. Each device contains a unique machine-readable identifier which is used to select the device for address assignment. The identifier is a binary bit string. Each bit position in the bit string is selected by the host in a serial manner with the host specifying which binary value is being solicited. All devices whose identifier digit matches the solicited value respond positively and remain in contention for address assignment. The other devices will not respond and drop out of contention for address assignment until the sequence is restarted from the first bit. After the bit sequence is completed, the address for that device is bused to the device, and the sequence is restarted from the first bit until all devices have been assigned an address.

EP 0 221 303 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 86 11 2765

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.4)
X,P	EP-A-0 173 905 (TEKTRONIX) * Figure 1; page 2, line 25 - page 4, line 11; page 7, lines 5-23; page 8, lines 7-10; page 8, line 2 - page 10, line 15; page 13, line 33 - page 14, line 2 *	1,10,12 ,13,15, 16	G 06 F 12/06
X	DE-A-3 347 357 (SIEMENS) * Figure 1; page 5, lines 1-10; page 6, lines 10-16; page 6, line 29 - page 7, line 8 *	12,13, 15	
Y		14	
A		1-11,16 ,17	
Y,D	US-A-4 360 870 (McVEY) * Figures 3,4; column 2, line 41 - column 3, line 2; column 3, lines 45-47; column 4, lines 28-49; column 6, lines 46-56 *	14	
X,P	EP-A-0 180 990 (SPACELABS) * Page 8, line 21 - page 9, line 25 *	12	TECHNICAL FIELDS SEARCHED (Int. Cl.4)
A	EP-A-0 044 949 (IBM) * Page 5, lines 1-27; page 10, line 23 - page 13, line 2 *	1-17	G 06 F 12/06 H 04 L 11/16 G 06 F 15/16

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	13-07-1989	MASCHE C.M.

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P0401)